

Analysis Report: news-api-project

peedaluk (GitHub)

June 30, 2025

1 Introduction

This report analyzes the **news-api-project** available at <https://github.com/peedaluk/news-api-project>. The project is a Flask-based web application that fetches and displays news articles using the NewsAPI service.

2 Project Structure

This image shows the organized file and folder structure of the news-api-project, including main application files, templates, static assets, and configuration files.



Figure 1: Project directory structure of news-api-project

3 Database Schema

This diagram illustrates the database schema, highlighting tables such as **search_logs**, their fields, and relationships relevant to the news search logging functionality.

```

PS C:\Users\koulade> mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 172
Server version: 9.3.0 MySQL Community Server - GPL
Copyright (c) 2000, 2025, Oracle and/or its affiliates.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE news_app;
Query OK, 1 row affected (0.009 sec)

mysql> USE news_app;
Database changed

mysql> CREATE TABLE search_logs (
  -> id INT AUTO INCREMENT PRIMARY KEY,
  -> query VARCHAR(255),
  -> fetched_at DATETIME DEFAULT CURRENT_TIMESTAMP
  -> );
Query OK, 0 rows affected (0.563 sec)

mysql>

```

Figure 2: Database schema used in the project

4 Backend Code Analysis (app.py)

Below is a partial code snippet from app.py, focusing on the main structure and logic of the Flask application:

Listing 1: Relevant blocks from app.py

```

from flask import Flask, render_template, request
from newsapi import NewsApiClient

app = Flask(__name__)
newsapi = NewsApiClient(api_key='YOUR_API_KEY')

@app.route('/', methods=['GET', 'POST'])
def index():
    articles = []
    query = ''
    if request.method == 'POST':
        query = request.form.get('keyword')
        if query:
            news = newsapi.get_everything(q=query, language='en', page_size=100)
            articles = news['articles']
            ...
    else:
        news = newsapi.get_top_headlines(language='en', page_size=100)
        articles = news['articles']
        ...
    return render_template('index.html', articles=articles, query=query)
...

```

5 Frontend (templates/index.html)

Below is a partial code snippet from index.html, showing the main div blocks for the search bar and news articles display:

Listing 2: Relevant div blocks from templates/index.html

```
<div class="container mt-4">
```

```

<h1 class="mb-4">Top Headlines</h1>
<form method="POST" class="mb-4">
  <div class="input-group">
    <input type="text" name="keyword" class="form-control" placeholder=
      "Search news by keyword..." value="{{ query }}">
    <button class="btn btn-primary" type="submit">Search</button>
  </div>
</form>
<div class="row">
  {% for article in articles %}
    <div class="col-md-4 mb-4">
      <div class="card h-100">
        ...
      </div>
    </div>
  {% endfor %}
</div>
</div>
...

```

6 Home Page

Screenshot of the application's home page displaying top news headlines with a search bar for keyword-based news retrieval.

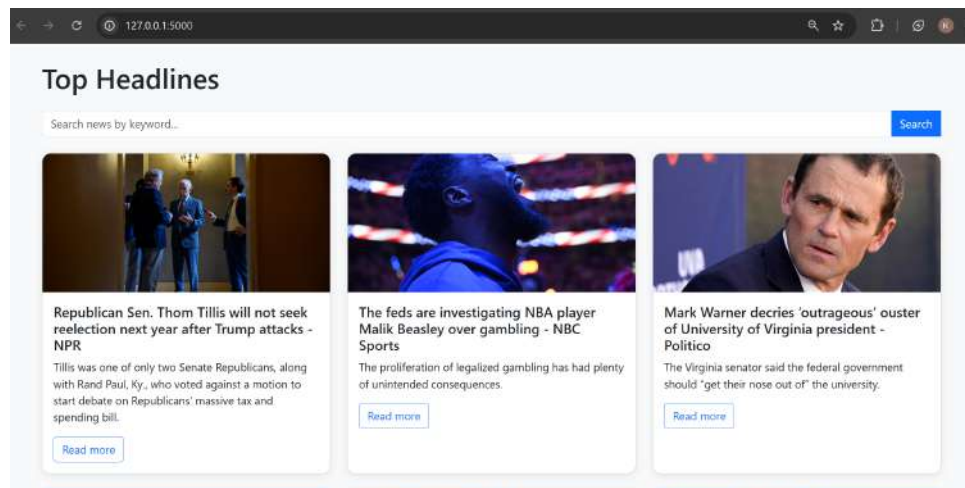


Figure 3: Home page of the News Web Application

7 Search Results

Screenshot showing the search results page where news articles are filtered and displayed based on the user's keyword input.

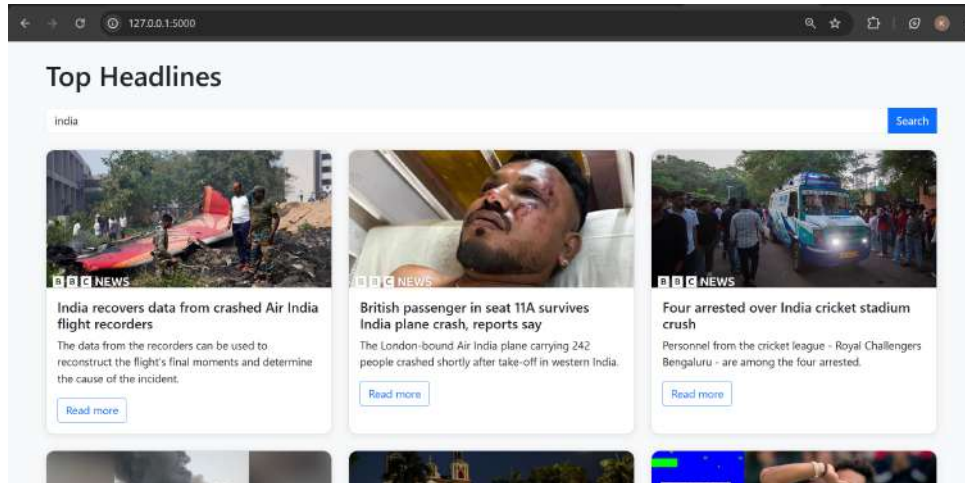


Figure 4: Search results for 'India' page displaying filtered news articles

8 Features

- Fetches and displays top news headlines.
- Supports keyword-based news search.
- Responsive web interface.

9 How It Works

1. User visits the homepage to see top headlines.
2. User can enter a keyword to search for specific news.
3. The backend fetches articles from NewsAPI and renders them.

10 Conclusion

The **news-api-project** is a clear, well-structured Flask web application for news aggregation. It demonstrates modular code organization and a user-friendly design. Future improvements could include user authentication, saving favorite articles, or supporting multiple news sources.

References

- peedaluk. (2023). *news-api-project* [Source code]. GitHub. <https://github.com/peedaluk/news-api-project>
- NewsAPI. <https://newsapi.org/>
- Flask Documentation. <https://flask.palletsprojects.com/>